An Improved Particle Swarm Optimization Algorithm based on Membrane Structure

Yanhua Zhong[1] , Shuzhi Nie[2]

[1]**Depart. Of Electronics and Information Technology,Jiangmen Polytechnic
Jiangmen, China**

[2]**Depart. Of Electronics and Information Technology,Jiangmen Polytechnic
Jiangmen, China**

Abstract
Presented a new hybrid particle swarm algorithm based on P systems, through analyzing the working principle and improved strategy of the elementary particle swarm algorithm. Used the particles algorithm combined with the membrane to form a community, particles use wheel-type structure to communicate the current best particle within the community. The best particles, as Representative, compete for the optimal particle of the higher level. Utilized the Objective Functions to test the designed algorithm performance, compared with other particle swarm optimization algorithms, the experiment results shown that the designed algorithm has better performance in seeking Optimization solution quality, robustness and convergence speed.
Keywords: *P system; Particle swarm; Hybrid intelligent algorithm*

1.Introduction

Particle swarm optimization (PSO) is a class of a stochastic optimization algorithm based on swarm intelligence. In PSO algorithm, each individual look as a particle in a d-dimensional search space, fly at a certain speed in the search space, dynamically adjust the speed according to themselves and companion flying experience. Each of the particles has an objective function to determine the adaptation value, particles search the optimal solution followed the current optimal particle in the solution spaces, find the optimal solution by iteration. In each iteration, particle by tracking individual extreme values and global extreme values to update myself, in the process of looking for the two extremes, the particle updates own speed and location according to the corresponding location [1].

PSO has easy to understand and implement, the global search ability and other characteristics, much broader field of science and engineering concern, has become the fastest growing intelligent optimization algorithms. As PSO's performance depends on the algorithm parameters, to overcome these shortcomings, national researchers have proposed various measures for improvement. These improvements aim is to improve the particle diversity, and enhance the global exploration ability of particle, or to improve local development capacity, and enhance the convergence speed and accuracy [2-4].

There are usually two kind of combination way: First, used other adaptive optimization shrinkage factor, inertia weight and acceleration constants; second, combined with PSO and other evolutionary algorithm operators or other technology. Juang C F. combined the ant algorithm and the PSO for solving discrete optimization problems; Robinson et al [4] and Juang the combination of GA and PSO are used to optimize the antenna design and recurrent neural network design [4-5]; Documents [6] divided the population into more sub-population, and then the different sub-populations using PSO or GA or independent evolution of hill-climbing; Naka et al [7] have studied that the selection operation of GAs introduced into the PSO according to select a certain rate of reproduction to copy better individuals; Angeline [8] introduced tournament selected into the PSO algorithm, according to the current location of individual fitness, to each individual and compared to several other individuals, and then compare the results to the whole group based on sorting, the best half of the particle in the current group replace the worst half of the position and speed of the position and speed, while preserving the memory of individuals of each individual the best position; El-Dib et al [9] proposed crossover operation on the particle position and speed; Higashi et al [10] introduced Gaussian mutation to PSO; Miranda [11] used the variation, selection and breeding a variety of operating at the same time update the formula in the neighborhood of the best locations, and inertia weight and acceleration constants; Zhang et al [12-14] using differential evolution operator to select the speed of update formula in the best position of particles; And Kannan et al [15] using differential evolution to optimize the PSO inertia weight and acceleration constants. Document [16] proposed a discrete quantum individual PSO; Document [17] proposed the quantum PSO that update particle position based on behavior of quantum.

In this paper, combine membrane computing ideas with standard PSO, form a membrane community solution space, the particle in the community will be quick convergence solution space effects. The best particles as Representative compete the optimal particle of the higher level. This can effectively prevent the fall into local minimum, premature convergence or stagnation.

2.Hybrid Algorithm Design

2. 1P System

Studied DNA computing for many years, inspired by biological cells, Gheorghe Păun proposed the membrane computing in 2000 [3], through dealing with compounds from layered structure of living cells to abstract the computing model. The model is called membrane systems or P system, A P system of n dimensional can be expressed as the following formula:

$$\Pi = (V, T, C, \mu, w_1, ..., w_m, (R_1, \rho_1), ..., (R_m, \rho_m)) \quad (1)$$

In formulary (1), V is alphabet, its elements are called objects; $T \subseteq V$ is output alphabet; $C \subseteq V - T$ is catalyst, its elements don't change in the evolutionary process, don't create new characters too, however, perform some evolution rules required its; μ is a membrane structure that contain m membranes, each membrane and its enclosed area show with a label set of H, $H = \{1, 2, \cdots, m\}$, then m is Π's dimensional; $w_i \in V^*$ ($1 \le i \le m$) show multiple sets that contain objects in the region i of membrane structure m, V^* is the collection of arbitrary character string composed of V's character; Evolution rules is a binary set of (u, v) , Usually written as $u \to v$, u is the string of V^*, $v = v'$ or $v = v'\delta$, v' is the character string of collection $\{a_{here}, a_{out}, a_{in_j} | a \in V, 1 \le j \le m\}$, δ is the special characters don't belong to V, When a rule contains δ, membrane is dissolved while the implementation of the rules, regard the length of u as the rules of $u \to v$ 'radius, R_i ($1 \le i \le m$) is the finite set of evolution rules, each R_i is associated with the region of i in the membrane structure of μ, ρ_i is the second order in R_i , which is the preferential relations, Which is the preferential relation that implementation the rules of n.

In short, P system consists of three parts: hierarchical structure of membrane; the multiple set of objects; evolution rules.

2.2Particle Swarm Optimization

Particle swarm optimization (PSO) is initialized to a group of random particles (random solutions), and then iterate to find the optimal solution. In each iteration, the particles by tracking the two extreme to update their own, the first one is the particle itself to find the optimal solution, this solution is called individual extreme; the other one is the whole population to find the optimal current solution, this extreme is a global extreme. In addition you can also use part of the particles rather than the entire population as a neighbor; the extreme in all neighbors is the local minimum.

Suppose a D-dimensional target in the search space, there are N particles to form a community in which the i-th particle is represented as a D-dimensional vector

$$X_i = (x_{i1}, x_{i2}, \cdots, x_{iD}), \quad i = 1, 2, \cdots, N \quad (2)$$

The first i particles "flying" velocity is also a D-dimensional vector, expressed as

$$V_i = (v_{i1}, v_{i2}, \cdots, v_{iD}), \quad i = 1, 2, \cdots 3 \quad (3)$$

Each particle also maintains a memory of its previous best position, represented as

$$p_{best} = (p_{i1}, p_{i2}, \cdots, p_{iD}), \quad i = 1, 2, \cdots, N \quad (4)$$

The optimal position can be search in whole group which can be represented as

$$g_{best} = (p_{g1}, p_{g2}, \cdots, p_{gD}) \quad (5)$$

When found these two optimal values, the particles update their speed and position according to the following formula (6) and (7).

$$v_{id} = w * v_{id} + c_1 r_1 (p_{id} - x_{id}) + c_2 r_2 (p_{gd} - x_{id}) \quad (6)$$

$$x_{id} = x_{id} + v_{id} \quad (7)$$

Where, c_1 , c_2 are learning factor, also known as the acceleration constant, r_1 , r_2 are random numbers in Range [0,1].

Equation (6) on the right consists of three parts, the first part of the "inertia (inertia)" or "momentum (momentum)" part, reflects the movement of particles "habit (habit)", the particle has the speed to maintain their previous trend; the second part of "cognitive (cognition)" part, reflects the historical experience of the particles on their own memory (memory) or memory (remembrance), the particle has the best location close to their historical trends; third part of "social (social) "part, reflects the particle collaboration and knowledge sharing between groups of historical experience, the particle with the group or neighborhood close to the historical best location trend.

According to experience, Usually take $c_1 = c_2 = 2$, $v_{id} \in [-v_{\max}, v_{\max}]$ is velocity of particle, r_1 , r_2

is random numbers in Range [0,1], $v_{\max}$ is a constant, set by the user to limit the speed of the particle.

2. 3Hybrid Algorithm

Particle swarm optimization based on Membrane computing, is assigned to different groups of particles in the membrane, a membrane is equivalent to the community, the membrane particles are selected the best of its specific particles and use the difference between him and the best neighbor vector instead of the standard PSO velocity update formula "part of individual consciousness." Topology of populations of membrane between each particle using wheel-type structure, can further improve the performance of the algorithm to avoid premature convergence phenomenon. In this structure, each particle with the best center particle exchange information in the membrane, the best particle in the community as representatives of the best particle in the higher level competition, the topology shown in Figure 1. With a knockout way to find this optimal solution to improve the speed of dissemination of information to avoid loss of information to better improve the efficiency of the algorithm.

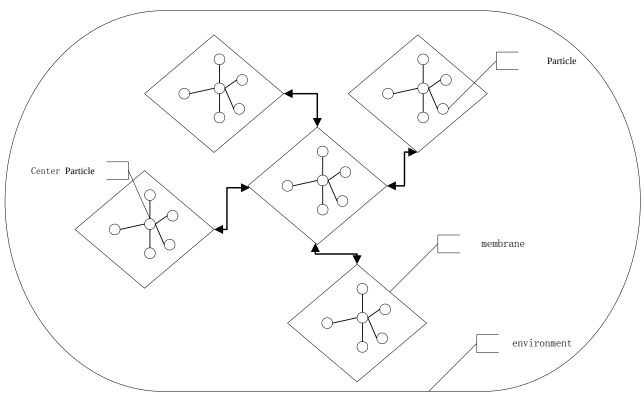

Fig. 1 Particle swarm based on P systems

Particle swarm optimization based on P system process is as follows:

- Step 1: the establishment of the structure contains m membranes, each membrane and the surrounding areas use label set $H = \{1, 2, \cdots j \cdots, m\}$.
- Step 2: The particles were randomly assigned to the m -membranes.
- Step 3: initialize the particle swarm, including population size N^j, the number of j representatives of membranes, each particle position x^j_i and velocity V^j_i.
- Step 4: to assess the fitness value $F^j_{it}[i]$ of each particle;
- Step 5: For each particle, compare its fitness value $F^j_{it}[i]$ and individual extremum p^j_{best}（i）, if $F^j_{it}[i] > p^j_{best}(i)$, Then replace p^j_{best}（i） with $F^j_{it}[i]$;
- Step 6:the best particle p^j_{best} in the membranes j, compare its fitness value $F^j_{it}[i]$ and Global extremum g_{best}, if $F^j_{it}[i] > p^j_{best}$, Then re place g_{best} with $F^j_{it}[i]$;;
- Step 7: According to formula (6), (7), update the particle velocity v^j_i and position x^j_i.

Step 8: If the end conditions are met to elect the best particle, the algorithm ends, otherwise return to Step 4.

3.Algorithm test and analysis

To verify the performance of the algorithm, we proposed in this paper based on particle swarm optimization membrane structure (mPSO) with standard PSO algorithm (sPSO) [1], differential evolution particle swarm optimization [12,13], GA particle swarm optimization [6] for comparison. In the experiment, we selected five standard optimization functions to test these algorithms, each function uses 10-dimensional search space.

3. 1Test functions

1) F1 De Jone's function

The simplest test function is De Jong's function 1. It is also known as sphere model. It is continuos, convex and unimdal. Function definition:

$$f_1(\bar{x}) = \sum_{i=1}^{D} x_i^2; x \in [-5.12, 5.12] \tag{8}$$

Global minimum: $f(\bar{0}) = 0;$

2) F2 Axis parallel hyper-ellipsoid

The axis parallel hyper-ellipsoid is similar to De Jong's function 1. It is also known as the weighted sphere model. Again, it is continuos, convex and unimodal. Function definition:

$$f_2(x) = \sum_{i=1}^{D} i x_i^2; x \in [-5.12, 5.12] \tag{9}$$

Global minimum: $f(0) = 0;$

3) F3 Sum of different Powers

The sum of different powers is a commonly used unimodal test function. Function definition:

$$f_3(\bar{x}) = \sum_{i=1}^{D} |x_i|^{(i+1)} \quad ; x_i \in [-1,1] \tag{10}$$

Global minimum: $f(0) = 0;$

4) *F4 Rastrigin's function*

Rastrigin's function is based on function 1 with the addition of cosine modulation to produce many local min-ima. Thus, the test function is highly multimodal. However, the location of the minima is regularly distributed. Function definition:

$$f_4(\bar{x}) = \sum_{i=1}^{D} 10 + x_i^{\ 2} - 10 \times \cos(2\pi x_i); x \in [-5.12, 5.12] \tag{11}$$

Global minimum: $f(0) = 0;$

5) *F5 Schwefel's function*

Schwefel's function is deceptive in that the global minimum is geometrically distant, over the parameter space, from the next best local minima. Therefore, the search algorithms are potentially prone to convergence in the wrong direction. Function definition:

$$f_5(\bar{x}) = \sum_{i=`}^{D} -x_i \times \sin\sqrt{|x_i|}; x \in [-500, 500] \tag{12}$$

Global minimum: $f(420.9687) = 418.9829;$

We use the standard initialization range, but the particular function to facilitate the display of its performance, its initialization to narrow the scope of [-1, 1]. For all of the function we use the 10-dimensional search space, Table I shows the five test function the initialization and the search space.

Table 1: Function initialization settings

test function	Search domain	Initialization range
F1 De Jone's function 1	[-5.12 , 5.12]	[-1 , 1]
F2 Sum of different Powers	[-1 , 1]	[-1 , 1]
F3 Rastrigin's function 6	[-5.12 , 5.12]	[-5.12,5.12]
F4Axis parallel yper-ellipsoid	[-5.12 , 5.12]	[-5.12,5.12]
F5 Schwefel's function 7	[-500,500]	[-500 ,500]

3.2Performance analysis

As the optimization result is random, for scientific comparison algorithms, we take the average of the results which independently test each test function 50 times. For each run, the end result of 1000 iterations to compare the accuracy of algorithm convergence. Can be found by observing, MPSO has better optimization ability for various functions, especially for multimodal function of search capability is much higher than other comparison algorithms. Table 4.2 shows Statistics for average optimal results of the algorithm which the five Objective functions are run 50 times independently.

Table 2: Convergence value of average

F	Di	N	Gma	Average			
	m		x	S PSO	DEPSO	GAPSO	MPSO
F1	10	50	1000	0.91232	9.3759e-023	0.0028096	6.8731e-005
F2	10	50	1000	1.0425e-007	7.1588e-022	1.0247	2.7028e-005
F3	10	50	1000	0.17698	2.7959e-036	3.3251e-007	9.0747e-008
F4	10	50	1000	90.653	43.751	14.467	0.23902
F5	10	50	1500	-3777.1	-2883.7	-2793.5	-4189.8

In the table, F is testing function ,N is the search space, N is running times, Gmax iterative times is 1000.

3.3Algorithm parameters analysis

Some of the parameter settings for MPSO, such as population size, may have an important effect on the performance of the algorithm, so experiment to verify the relationship between algorithm parameters and performance. The function F5 was be selected as the evaluation function to test the algorithm convergence precision under different parameter settings. Assuming population 30, particle number 30 in one membrane, Exchange particle 20 at a time, switching time 40 generations, the following is assuming other conditions remain unchanged following the case of change in one parameter experimental results are as follows:

- Number of particles within populations, population sizes was taken 30, 40, and 50.
- Population size (number of membranes), were taken twenty, thirty, forty populations.
- Exchange particle time, exchange time were taken 30 generations, 40generations, 50 generations.
- The number of particles each exchange, the number of exchange particles were taken 15, 20, 25.

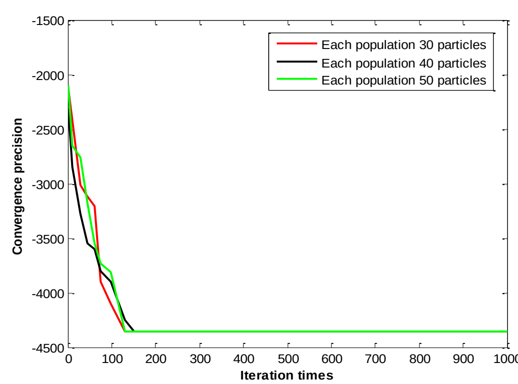

Fig. 2 Number of particles within populations

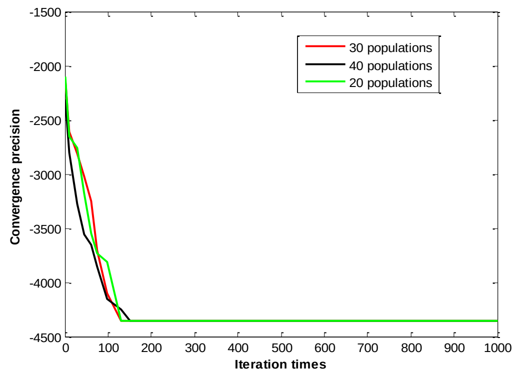

Fig. 3 Population size

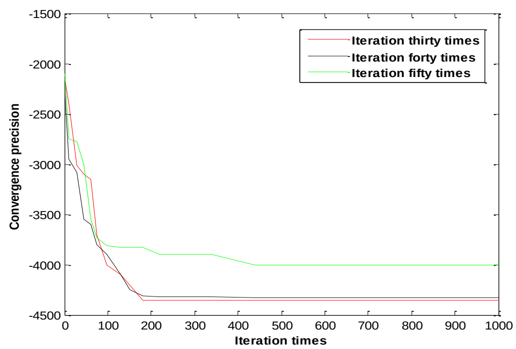

Fig. 4 Exchange particle time

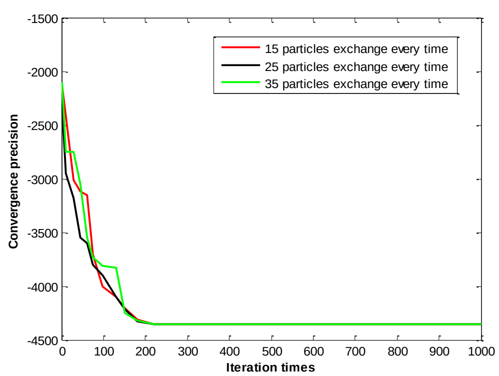

Fig. 5 The number of particles each exchange

By several key parameters for mPSO to be set different values, experimental results as follows: As show in the Figure 2 that the number of particles within populations has little effect on convergence accuracy. In each set 40 particles within a population have the best convergence results. Figure 3 shows the number of membrane was set in the experiment, the number of membrane were set to (20, 30, 40), the populations that was set 40 has obvious advantages.

As seen in Figure 4, the results of different exchange timing which were to take timing (30, 40, 50) generation. When iteration take to the 30 that the results is best, However, The algorithm cannot converge to the global minimum when the exchange timing was set 50. The number of particles each exchange are shown in Figure 5, When the particles were set in (15, 20, 25), the optimization effect is had little fluctuation, but each time exchange 15 particles is better.

4.Conclusion

In this paper, proposed a new hybrid particle swarm algorithm based on P system, which is through the analysis of elementary particle swarm algorithm the working principle and improvement a strategy. And the algorithm is excellent in finding extreme for multimodal function. But there is still much scope for improvement in this algorithm, population exchange between the particles are too simple, stupid, and tend to a single. Future improvements can be used with the guidance of the dynamic evolution of membranes exchange, according to the algorithm to optimize conditions for specific particle exchange operation. Hybrid evolutionary algorithm is one trend in the field of evolutionary algorithms, so that the advantages of different algorithms can be integrated and "better, faster, cheaper" to get the solution of the problem, it is a valuable research direction.

References

[1] Kennedy J, Eberhart R. Particle swarm optimization. in: Proceedings of the 4th IEEE International Conference on Neural Networks, Piscataway: IEEE Service Center, 1995, pp.1942 -1948.
[2] Zhang G. X,Liu C. X,Gheorghe M.,P ăé F. Solving satisfiability problems with membrane algorithms. Proceedings of the Fourth International Conference on Bio-Inspired Computing: Theories and Applieations.2009:29-36.
[3] Fen Zhou, et al, .A Particle Swarm Optimization Based on P systems. 2010 Sixth International Conference on Natural Computation (ICNC 2010) .pp.3003-3007.
[4] Gh. Pˇ aun. Computing with membranes. Journal of Computer and System Sciences.2000, 1(61):108-143.
[5] Juang C F. A Hybrid of Genetic Algorithm and Particle Swarm Optimization for Recurrent Network Design. IEEE Transactions on Systems, Man and Cybernetics (Part B-Cybernetics), 2004, 34 (2): 997-1006.
[6] Krink T and Løvbjerg M, The Lifecycle Model: Combining Particle Swarm Optimization, Genetic Algorithms and Hill Climbers. In Proc. of PPSN 2002, pp. 621-630.
[7] S. Naka, T. Genji, T.Yura, and Y. Fukuyama. A hybrid particle swarm optimization for distribution state estimation. IEEE Trans. Power Syst, pp. 60-68, Feb. 2003.
[8] Angeline P.J.Using Selection to Improve Particle Swarm Optimization. In Proc of the IEEE Congress on Evolutionary Computation. Anchorage, USA, 1998,84-89.
[9] El-Dib, H. Youssef, M. El-Metwally, and Z. Osman. Load flow solution using hybrid particle swarm optimization. In

Proc. Int. Conf.Elect., Electron. Comput. Eng., Sep. 2004, pp. 742-746.

[10] Higashi N, Iba H. Particle Swarm Optimization with Gaussian Mutation. in Proc. of the 2003 Congress on Evolutionary Computation. 2003. 72289.

[11] V. Miranda and N. Fonseca. EPSO: evolutionary particle swarm optimization, a new algorithm with applications in power systems. in IEEE/PES Transmission and Distribution Conf. Exhibition Asia Pacific, Oct. 2002, vol. 2, pp. 745-750.

[12] W. Zhang and X. Xie. DEPSO: Hybrid particle swarm with differential evolution operator. in Proc. IEEE Int. Conf. Syst., Man, Cybern., Oct. 2003, vol. 4, pp. 3816-3821.

[13] H. Talbi and M. Batouche, Hybrid particle swarm with differential evolution for multimodal image registration. in Proc. IEEE Int. Conf. Ind. Technol., Dec. 2004, vol. 3, pp. 1567-1572.

[14] P. Moore and G. Venayagamoorthy. Evolving combinational logic circuits using particle swarm, differential evolution and hybrid DEPSO. Int. J. Neural Syst., vol. 16, no. 2, pp. 163-177, 2006.

[15] S. Kannan, S. Slochanal, and N. Padhy. Application of particle swarm optimization technique and its variants to generation expansion problem. ELSERVIER Electric Power Syst. Res., vol. 70, no. 3, pp.203-210, Aug. 2004.

[16] [S. Yang, M. Wang, and L. Jiao. A quantum particle swarm optimization. in roc. IEEE Congr. Evol. Comput., Jun. 2004, vol. 1, pp.320-324.

[17] J. Sun, Particle Swarm Optimization with Particles Having Quantum Behavior. in Proc. 2004 Congress on Evolutionary Computation, pp. 325-331.